UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/938,029	08/23/2001	Shawfu Chen	POU920000197US1	1671

46369 7590 11/12/2004

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
MANOSKEY, JOSEPH D	

ART UNIT	PAPER NUMBER
2113	

DATE MAILED: 11/12/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/938,029	CHEN ET AL.	
	Examiner	Art Unit	
	Joseph Manoskey	2113	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 August 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-65* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-9,13-29,33-53 and 57-65* is/are rejected.

7)☒ Claim(s) *10-12,30-32 and 54-56* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 August 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claim 1-6, 8, 9, 13-18, 20-26, 28, 29, 33-38, 40-50, 52, 53, 57-62, 64, and 65 are rejected under 35 U.S.C. 102(b) as being anticipated by Brent et al., U.S. Patent 5,459,864, hereinafter referred to as "Brent".

3. Referring to claim 1, Brent teaches load balancing and automatic recovery of queue processors, this is interpreted as a method of switching queue ownership (See Col. 1, lines 18-20). Brent teaches one processor detecting a failure signal from the other processor, this is interpreted as obtaining an indication that a queue is to be taken over, said queue being resident in memory of a first processor (See Col. 5, lines 39-44). Brent discloses the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted as moving the queue from the first processor to a second processor, the queue to be resident in memory of the second processor and not resident in the memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

4. Referring to claim 2, Brent teaches the first processor being failed and stopped, this is interpreted as the first processor being inactive (See Col. 5, lines 49-52).

5. Referring to claim 3, Brent discloses the use of checkpointing and recovery features, this is interpreted as rebuilding the queue prior to moving said queue (See Col. 5, lines 35-36).

6. Referring to claim 4, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

7. Referring to claim 5, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

8. Referring to claim 6, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing the rebuilt version of the queue to the checkpoint of the queue (See Col. 6, lines 4-7).

9. Referring to claim 8, Brent discloses using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as each in-doubt event of the one or more in-doubt events represents a most recent event for a message of the queue (See Col. 5, line 53 to Col. 6, line 3).

10. Referring to claim 9, Brent discloses the work being redistributed to any other processor, this is interpreted as moving comprising reading the queue into the memory of the second processor (See Col. 5, lines 49-52).

11. Referring to claim 13, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

12. Referring to claim 14, Brent discloses load balancing and automatic recovery of queue processors, this is interpreted as a method of reconstructing queues (See Col. 1, lines 18-20). Brent discloses the use of checkpointing and recovery features, this is

interpreted as rebuilding contents of a queue to obtain an updated version of the queue, the queue being in memory resident queue of a first processor (See Col. 5, lines 35-36). Finally, Brent teaches the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted reading at least a portion of the updated version of the queue into memory of a second processor, the second processor being different than said first processor, and wherein said at least a portion of the updated version of the queue on longer resides in local memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

13. Referring to claims 15 and 16, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

14. Referring to claim 17, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as the version of the queue being deleted is an updated version written to the checkpoint of the first processor, in response to the rebuilding (See Col. 6, lines 4-7).

15. Referring to claim 18, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

16. Referring to claim 20, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

17. Referring to claim 21, Brent teaches load balancing and automatic recovery of queue processors, this is interpreted as a system of switching queue ownership (See Col. 1, lines 18-20). Brent teaches one processor detecting a failure signal from the other processor, this is interpreted as obtaining an indication that a queue is to be taken over, said queue being resident in memory of a first processor (See Col. 5, lines 39-44). Brent discloses the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted as moving the queue from the first processor to a second processor, the

queue to be resident in memory of the second processor and not resident in the memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

18. Referring to claim 22, Brent teaches the first processor being failed and stopped, this is interpreted as the first processor being inactive (See Col. 5, lines 49-52).

19. Referring to claim 23, Brent discloses the use of checkpointing and recovery features, this is interpreted as rebuilding the queue prior to moving said queue (See Col. 5, lines 35-36).

20. Referring to claim 24, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

21. Referring to claim 25, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

22. Referring to claim 26, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing the rebuilt version of the queue to the checkpoint of the queue (See Col. 6, lines 4-7).

23. Referring to claim 28, Brent discloses using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as each in-doubt event of the one or more in-doubt events represents a most recent event for a message of the queue (See Col. 5, line 53 to Col. 6, line 3).

24. Referring to claim 29, Brent discloses the work being redistributed to any other processor, this is interpreted as moving comprising reading the queue into the memory of the second processor (See Col. 5, lines 49-52).

25. Referring to claim 33, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

26. Referring to claim 34, Brent discloses load balancing and automatic recovery of queue processors, this is interpreted as a system of reconstructing queues (See Col. 1,

lines 18-20). Brent discloses the use of checkpointing and recovery features, this is interpreted as rebuilding contents of a queue to obtain an updated version of the queue, the queue being in memory resident queue of a first processor (See Col. 5, lines 35-36). Finally, Brent teaches the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted reading at least a portion of the updated version of the queue into memory of a second processor, the second processor being different than said first processor, and wherein said at least a portion of the updated version of the queue on longer resides in local memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

27. Referring to claims 35 and 36, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

28. Referring to claim 37, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as the version of the queue being deleted is an updated version written to the checkpoint of the first processor, in response to the rebuilding (See Col. 6, lines 4-7).

29. Referring to claim 38, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

30. Referring to claim 40, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

31. Referring to claim 41, Brent discloses load balancing and automatic recovery of queue processors, this is interpreted as a system of switching queue ownership (See Col. 1, lines 18-20). Brent teaches one processor detecting a failure signal from the other processor, this is interpreted as a queue being resident in memory of a first processor (See Col. 5, lines 39-44). Brent discloses the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted as at least a second processor adapted to move the queue from the first processor to the second processor, the queue to be

resident in memory of the second processor and not resident in memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

32. Referring to claim 42, Brent teaches load balancing and automatic recovery of queue processors, this is interpreted as a system of switching queue ownership (See Col. 1, lines 18-20). Brent teaches one processor detecting a failure signal from the other processor (See Col. 5, lines 39-44). Brent discloses the use of checkpointing and recovery features, this is interpreted as a first processor adapted to rebuild contents of a queue to obtain an updated version of the queue being memory resident queue of a second processor (See Col. 5, lines 35-36). Brent discloses the use of plural processors, this is interpreted as three processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted as a third processor adapted to read at least a portion of the updated version of the queue into memory of the third processor, said third processor being different than said second processor, and wherein said at least a portion of the updated version of the queue no longer resides in local memory of the second processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

33. Referring to claim 43, Brent discloses resetting the first processor to enable to attempt to re-execute, this is interpreted as the first and third processor being the same (See Col. 5, lines 44-49).

34. Referring to claim 44, Brent teaches the system having plural processors and having a recovery features, this is interpreted as the first and third processor being different processors (See Col. 1, lines 18-21 and Col. 5, lines 35-36).

35. Referring to claim 45, Brent teaches load balancing and automatic recovery of queue processors, this is interpreted as a program storage device that is readable by a computing unit containing instructions for a method of switching queue ownership (See Col. 1, lines 18-20). Brent teaches one processor detecting a failure signal from the other processor, this is interpreted as obtaining an indication that a queue is to be taken over, said queue being resident in memory of a first processor (See Col. 5, lines 39-44). Brent discloses the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted as moving the queue from the first processor to a second processor, the queue to be resident in memory of the second processor and not resident in the memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

36. Referring to claim 46, Brent teaches the first processor being failed and stopped, this is interpreted as the first processor being inactive (See Col. 5, lines 49-52).

37. Referring to claim 47, Brent discloses the use of checkpointing and recovery features, this is interpreted as rebuilding the queue prior to moving said queue (See Col. 5, lines 35-36).

38. Referring to claim 48, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

39. Referring to claim 49, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

40. Referring to claim 50, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing the rebuilt version of the queue to the checkpoint of the queue (See Col. 6, lines 4-7).

41. Referring to claim 52, Brent discloses using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as each in-doubt event of the one or more in-doubt events represents a most recent event for a message of the queue (See Col. 5, line 53 to Col. 6, line 3).

42. Referring to claim 53, Brent discloses the work being redistributed to any other processor, this is interpreted as moving comprising reading the queue into the memory of the second processor (See Col. 5, lines 49-52).

43. Referring to claim 57, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

44. Referring to claim 58, Brent discloses load balancing and automatic recovery of queue processors, this is interpreted as a program storage device that is readable by a computing unit containing instructions for a method of reconstructing queues (See Col. 1, lines 18-20). Brent discloses the use of checkpointing and recovery features, this is interpreted as rebuilding contents of a queue to obtain an updated version of the queue, the queue being in memory resident queue of a first processor (See Col. 5, lines 35-36). Finally, Brent teaches the use of plural processors, this is interpreted as two processors, and that the work of the failed processor is redistributed to the other processors, this is interpreted reading at least a portion of the updated version of the queue into memory of a second processor, the second processor being different than said first processor, and

wherein said at least a portion of the updated version of the queue on longer resides in local memory of the first processor (See Col. 1, lines 18-21 and Col. 5, lines 49-52).

45. Referring to claims 59 and 60, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as writing a version of the queue to a checkpoint of the second processor (See Col. 6, lines 4-7). Brent also teaches the failing processor being stopped and removed from the operational subsystem, this is interpreted as deleting a version of the queue from a checkpoint of the first processor (See Col. 5, lines 49-52).

46. Referring to claim 61, Brent teaches the recovery feature enabling full recovery of the operations, this is interpreted as the version of the queue being deleted is an updated version written to the checkpoint of the first processor, in response to the rebuilding (See Col. 6, lines 4-7).

47. Referring to claim 62, Brent teaches using the checkpoint to re-establish execution of incomplete requests on any processor in the subsystem, this is interpreted as processing at least a portion of a recovery log of said first processor to obtain one or more in-doubt events for the queue and merging at least on in-doubt event of the one or more in-doubt events with a checkpoint of the queue to obtain a rebuilt version of the queue, the checkpoint being associated with the first processor (See Col. 5, line 53 to Col. 6, line 3).

48. Referring to claim 64, Brent teaches the use of checkpointing and recovery features, this is interpreted as rebuilding comprising a recovery log and a checkpoint of the queue to rebuild contents of the queue, the recovery log and the checkpoint being associated with the first processor (See Col. 5, lines 33-36).

49. Referring to claim 65, Brent teaches the first processor being failed and stopped, this is interpreted as wherein moving is performed in response to an indication that the queue is to be taken over and not in response to performing a transaction in which the queue is processed (See Col. 5, lines 49-52).

Claim Rejections - 35 USC § 103

50. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

51. Claim 7, 19, 27, 39, 51, and 63 are rejected under 35 U.S.C. 103(a) as being unpatentable over Brent in view of Dievendorff et al., European Patent Application 0280773 A2, hereinafter referred to as "Dievendorff".

52. Referring to claims 7, 19, 27, 39, 51, and 63, Brent discloses all the limitations (See rejection of claims 5, 18, 25, 38, 49, and 62 respectively) except processing a portion of the recovery log in reverse order, however Brent does disclose the use of checkpoints and recovery features (See Col. 5, lines 35-36). Dievendorff teaches the redo records being clustered towards the end of the log (See page 3, lines 20-21), this is interpreted as the entries that are used to rebuild a checkpointed queue and being done in reverse order since they are at the end of the log. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the checkpoint and recovery features of Brent with the reading in reverse of Dievendorff. This would have been obvious to one in the art at the time of the invention to do because it reduces the span of the log which must be read during the queue reconstruction (See Dievendorff, page 3, lines 23-24).

Allowable Subject Matter

53. Claims 10-12, 30-32, and 54-56 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Response to Arguments

54. Applicant's arguments, see pages 15-18 of amendment filed on 16 August 2004, with respect to the rejection(s)of claim(s) 1-64 under 35 U.S.C. 102(b) and 35 U.S.C. 103(a) have been fully considered and are persuasive. Therefore, the rejection has

been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Brent, see above rejection.

Conclusion

55. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joseph Manoskey whose telephone number is (571) 272-3648. The examiner can normally be reached on Mon.-Fri. (7:30am to 4pm).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoliel can be reached on (571) 272-3645. The fax phone

number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JDM
November 4, 2004

ROBERT BEAUSOLIEL
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100